Watching inside the Screen: Digital Activity Monitoring for Task Recognition and Proactive Information Retrieval

Vuong, Tung

2017-09-11

Vuong , T , Jacucci , G & Ruotsalo , T 2017 , ' Watching inside the Screen: Digital Activity Monitoring for Task Recognition and Proactive Information Retrieval ' , Proceedings of ACM on interactive, mobile, wearable and ubiquitous technologies , vol. 1 , no. 3 , 109 . https://doi.org/10.1145/3130974

http://hdl.handle.net/10138/308882
https://doi.org/10.1145/3130974

acceptedVersion

Watching inside the Screen: Digital Activity Monitoring for Task Recognition and Proactive Information Retrieval

TUNG VUONG, University of Helsinki
GIULIO JACUCCI, University of Helsinki
TUUKKA RUOTSALO, University of Helsinki

We investigate to what extent it is possible to infer a user's work tasks by digital activity monitoring and use the task models for proactive information retrieval. Ten participants volunteered for the study, in which their computer screen was monitored and related logs were recorded for 14 days. Corresponding diary entries were collected to provide ground truth to the task detection method. We report two experiments using this data. The *unsupervised task detection experiment* was conducted to detect tasks using unsupervised topic modeling. The results show an average task detection accuracy of more than 70% by using rich screen monitoring data. The *single-trial task detection and retrieval experiment* utilized unseen user inputs in order to detect related work tasks and retrieve task-relevant information on-line. We report an average task detection accuracy of 95%, and the corresponding model-based document retrieval with Normalized Discounted Cumulative Gain of 98%. We discuss and provide insights regarding the types of digital tasks occurring in the data, the accuracy of task detection on different task types, and the role of using different data input such as application names, extracted keywords, and bag-of-words representations in the task detection process. We also discuss the implications of our results for ubiquitous user modeling and privacy.

CCS Concepts: •**Information systems →Users and interactive retrieval;**

Additional Key Words and Phrases: Activity recognition, task detection, digital activity monitoring, screen scraping, user modeling

ACM Reference format:
Tung Vuong, Giulio Jacucci, and Tuukka Ruotsalo. 2017. Watching inside the Screen: Digital Activity Monitoring for Task Recognition and Proactive Information Retrieval. *PACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 109 (September 2017), 23 pages.
DOI: http://doi.org/10.1145/3130974

1 INTRODUCTION

Activity recognition is an important area for ubiquitous computing traditionally approached through analysing noisy low-level data gathered by sensors to discover and extract patterns that could be interpreted as meaningful activities [20]. Currently, many of our real-world activities are mediated, not only by sensors, but also by our behaviour and activity through digital interactions with a variety of computing services; the emails we read and send, the documents we write or read, or the applications that we use in many specialised tasks. Consequently, addressing activity or task recognition through analysing broader digital traces can reveal the tasks that are meaningful for users and can be useful in different respects; for example, to proactively recommend information.

This research was partially funded by TEKES (Re:Know) and the Academy of Finland (278090, 305739).

DOI: http://doi.org/10.1145/3130974

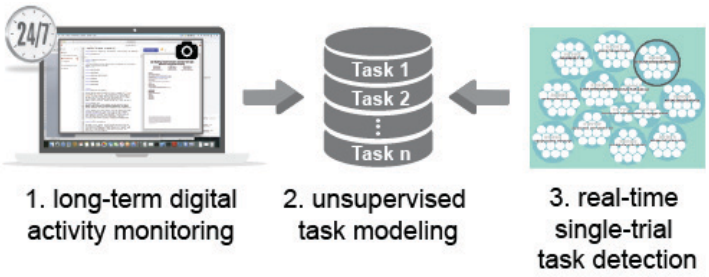


Fig. 1. A user model targeted to detect participants' digital tasks are modeled using an unsupervised method from digital activity monitoring data. Correct tasks are identified with greater than 70% accuracy, a single-trial task detection and proactive information retrieval from unseen interaction data with an accuracy of 95%, and the corresponding model-based document retrieval with Normalised Discounted Cumulative Gain of 98%.

The tasks that users perform in the digital world often comprise of a heterogeneous collection of activities necessary to accomplish these tasks. Users' information needs also often depend on their tasks, and effective information retrieval must be based on an understanding of users' task contexts [2]. For example, a user working in human-resource management on recruiting a summer trainee would need to read recruitment policy instructions, write a job advertisement, answer to emails, and go through job applications, to name a few activities related to the overall task spanning across a variety of applications and information items.

To detect tasks and retrieve associated information items, prior work has focused on developing user models and has made inferences about the user's information needs [7, 27]. Conventionally, user models and the associated data acquisition have been confined to pre-formatted interaction logs from certain applications or a set of services. A good example of logs used as the basis for user models are clicks of document links on search engine result pages, that are in turn used to rank the search results to better fit the user's needs [19]. While such models have been shown to be useful, they are only scratching the surface of all possible data that could be observed from a user's digital activity.

We propose a method of task modeling from natural computer usage via digital activity monitoring inside the screen. Computer screens can reveal rich information about the communication between the user and the computer. It is manifested in the applications we use, the content we examine or produce, and the entities, such as other people, with whom we relate and communicate. In our approach, screen recording software watches all information that is presented for the user. The resulting data representing the digital activity is then fed into an unsupervised machine-learning method in order to build a user model to detect the tasks that the user was engaged with.

The overall aim of the research is to model digital tasks from screen captures for task-aware information retrieval. To achieve that, we set out to solve two problems: First, understand to what extent it is possible to identify, label, and recover users' tasks and related information from screen captures; second, to investigate how a learned user model from screen captures can be used to proactively retrieve information. The challenge compared to previous approaches is that we do not confine the learning data to structured information from specific systems nor have separate user input, but attempt to learn from any frame displayed to a user from any application. For example, a conventional user modeling approach in a search engine would observe user input (e.g. a click) targeted to a data structure (e.g. specific document ID), and could use this data in the modeling process. Conversely, we study what can be learned from the user without any control over the input or data structure, but only by monitoring the user's digital activities inside the screen.

To study what can be inferred from digital activity monitoring and to what extent it is possible to detect users' tasks, we ask the following research questions:

- How accurately can we detect tasks and activities of the user from digital activity monitoring data using unsupervised learning?
- How accurately can we use the resulting task model for on-line task detection and proactive retrieval of task-relevant information?

Both questions will be addressed in each of two experiments: 1) a task classification and retrieval experiment and 2) a single-trial task detection and retrieval experiment using previously unseen information.

- *A task detection experiment* comprises of two parts. The first part is *screen monitoring*, during which all screen frames of digital activities from volunteers' laptops are recorded and corresponding diary entries of tasks are collected. We further classify tasks into categories to understand how many types of tasks occurred during 14 days of monitoring data and the accuracy of task detection with different task types. The second part is an *unsupervised task detection*, during which unsupervised machine learning is used to automatically detect, label, and retrieve information related to the tasks are compared to participants' assessments and diary entries.
- *A single-trial task detection and retrieval experiment*, during which the method's effectiveness in detecting tasks and retrieving task-relevant information in response to previously unseen interactions is studied.

Furthermore, we report the effectiveness of different data sources, such as application context, document content, and named entities. We conclude by drawing implications of our findings for user modeling and privacy.

To our knowledge, this is the first attempt to model digital tasks and proactively retrieve task-relevant information by capturing natural user behaviour by using all user activity recorded via a computer screen. We also demonstrate an application of the task detection and show that the learned model can be effectively used for proactive information retrieval.

2 BACKGROUND

The present work is related to several research areas, including user modeling, task detection, and long-term user activity monitoring. In the following subsections, we review prior work in each of these areas.

2.1 User Modeling

User modeling is the process of creating a conceptual understanding of a user to adapt systems to the user's specific needs [12]. Monitoring digital user traces and detecting tasks with user signals is part of user modeling research with the overwhelming target of understanding users to reduce the interaction necessary to operate systems and services. User modeling is not only tied to technical solutions to collect information and make inferences about users, but also to understand what can be learned from different types of digital traces and to what extent this benefits users.

Conventionally, user models are built from *explicit behavioural data* [12], such as the queries users submit, the links or menu options clicked, or items browsed, and they are specific to certain applications, such as news browsing [3], personal information management tools [15], or search rankings [32]. Such observations of explicit behavioural data are usually confined to a system or a set of services and can harness the data model which is structured based on prior knowledge about the content.

By far, the most popular application area for user modeling is search engines where user models are often built from implicit behavioural data to optimise rankings [32] or a mixture of implicit and explicit behaviour, including previous queries and click through information [26, 28].

2.2 Task detection

Recent research has identified the importance of detecting users' tasks to model a wider range of user's interests from limited input [16]. The task models generalize limited user input to a model that represents the broader interests of the user. Task models have an intuitive appeal: A user's digital activities and information that the user consumes involve a type of routine, and always serve a goal. The process of fulfilling the goal is some primary task that the user conducts outside of the digital environment [17].

The main user signals that have been used to detect tasks are search engine query logs [29] and the corresponding application target for these models has been personalizing information retrieval systems [33]. Query logs are often used for within-session learning to infer the short-term preferences of a particular user [11]. Recent research has extended this work to "task-based sessions"; sets of possibly non-contiguous queries within a session that correspond to a user task [22, 23]. Similar approaches also have been experimented within entity-based data representation [35].

All of these studies have concentrated on limited user input, typically search engine interaction data, and relatively short sessions varying from seconds to hours. Some attempts to model and make inferences about long-term user preferences have also been proposed, but also with limited input data including clicks and queries [31].

2.3 Long-term Activity Monitoring

A recent research study coming closest to ours, is a user modeling system for personal assistants, in which researchers at Google looked at several months of user history to identify not just short-term tasks, but also long-term interests and habits [13]. However, as with most previous work, this data was limited to search engine logs, in contrast to our approach of having a complete 24/7 view of the participants' digital activity. Another recent study in activity recognition was based on social media data, which exploited social media posts and location-based services to make inference about human behaviour [37, 38]. Nevertheless, social media data is still very limited as it only detects the majority of activities in entertainment and socializing, whereas our work has more general coverage regarding task types that also include work and important tasks, and is able to capture activities that go beyond individual services.

In general, the limited scope of investigating individual services or data sources involves a risk of deficient or information-poor user models. Research has suggested that information-poor user models can be harmful and waste our attentional resources, distracting us with irrelevant content. For example, providing recommendations for things of which we already are aware or that are not related to our present context, can cause users to feel bemused at best, frustrated at worst. Users have a low threshold for how many poor experiences they are willing to endure before a service loses its allure [9].

Researchers have recently introduced the concept of "everyday surveillance" [5], referring to the opportunities and threats posed by the possibility of extensive surveillance and personal data collection. However, there is a diminishing amount of instrumented experimental research collecting data and making inferences to push the boundaries of what everyday surveillance means and what can be learned from data collected through using digital surveillance. Conversely, most of the previous work has concentrated on qualitative analysis, surveys, or other measurements of users' attitudes and opinions [14].

The more data we can collect from users, the better we can understand the users and their needs. However, as indicated, previous research has been tied to specific applications and user signals captured in these applications, such as search engines. As a consequence, there is surprisingly little research regarding user modeling to maximize the kind of data that could be used in general-purpose user models [21].

In summary, previous work has focused on user modeling by 1) using application-specific interaction logs, such as queries submitted to search engines and results clicked, and typically 2) utilized in short-term user modeling –

manifested as sessions durations varying from seconds to hours, at best. To our knowledge, our research is the first to study what can be learned from data of digital activity monitoring data and to what extent these data can inform us about user's interests and tasks.

3 TASK DETECTION EXPERIMENT

The purpose of the task detection experiment was twofold. First, to understand the kind of tasks participants performed within the 14 days during the digital monitoring period. Second, to investigate the accuracy of task detection by applying an unsupervised machine learning method to the data from digital activity monitoring.

3.1 Tasks and Activities

Digital activities are parts of complex tasks that users perform using digital information resources and tools, which are here called simply tasks. In our case, tasks are performed using information resources, such as documents, applications, or other computing services. Tasks are characterized by their digital nature meaning that information processing plays a key role in these tasks. Previous research has manifested that the boundary between tasks, and tasks and activities is blurred [6]. Therefore, the present work focuses on tasks as concrete processes of the users that are labeled by the users. That is, a task is a complex collection of activities that span in time but that are meaningful for a user and can be meaningfully labeled by the user.

A digital task can be viewed as a concrete set of sequencing activities that share a common topical context, and the task's goal that spans across a variety of applications spanning over time and involving routine. For example, a task could be a wider ongoing project related to many documents read or written by the user with different applications, or digital communication with other people. Alternatively, a task could be a smaller daily routine task, such as daily news reading. Figure 2 demonstrates a realistic example of a wider task of "recruiting a summer trainee" from one participant. It shows how the task spans through applications and content from email and web documents to text documents edited via a word processing software, but that is still coherent for the user, and all information and activities within the task are associated with the recruiting process. Our focus is in macro tasks for which the user can provide a meaningful label. Macro task can be composed of many sub-tasks or micro tasks that serve the overall goal of the macro task.

3.2 Screen Monitoring

Digital monitoring of real-life computer usage for real-life digital tasks was a prerequisite for the experiments. We aimed at a methodology and the corresponding technology setup able to constantly and unobtrusively collect all possible digital activities of participants' digital behaviour.

Capturing the information on a computer screen holds great potential for capturing all possible digital information as it allows access to all visually communicated input and output (i.e all user generated visual content and visual content presented to the user on the screen). We therefore decided to use screen recording to capture all visual traces. Unlike other logging methods, screen recording has no restraints in terms of application range or user input, and apart from audio, it can capture every input and presentation of content that occurs between the human and the computer.

3.2.1 Apparatus. We use a screen capture logger to record images of active windows at five second intervals or every frame that indicates information change on the screen. Screen frames that are staying idle and constant window switch behaviour are not collected to compress allocated CPU and memory of the logger. In addition, we recorded operating system information, such as the name of the active window and timestamps.

The digital activity monitoring system is comprised of four components: screen capture (SC) logger, Optical Character Recognition (OCR) system, keyword extraction (KE) system, and Operating System (OS) logger.

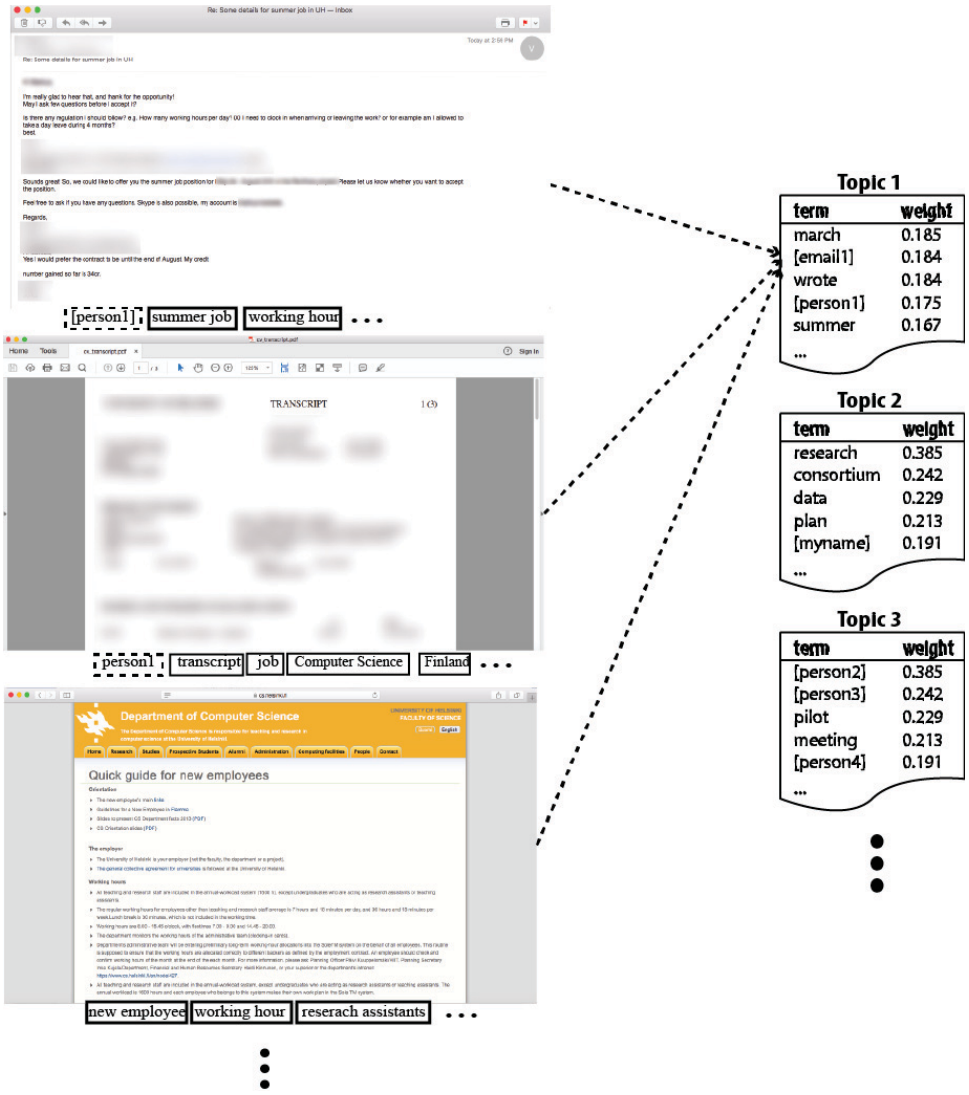


Fig. 2. An example of a task was modelled based on latent semantic structure of the text body of screen captures. In the figure, the *"recruitment of a summer trainee"* task is consisted of activities that span across several applications (email, word processing, Web browser).

- *SC logger* was developed in two versions. A Mac OS version was implemented by using the Core Graphics API and a MS Windows OS version was implemented by using the Desktop App UI, both of which are native operating system libraries. They performed an identical recording of the active window on users' screens and saved the captured screens as images. The SC also tracked mouse movements and keystrokes. Any change in the mouse position or a keystroke caused the logger to activate and wait for a 5-second interval until no further input was observed and commence recording the active window snapshot.

Therefore, duplicate screenshots were avoided, memory overload due to constant screen capturing after non-informative change on the screen was avoided, and screenshots were not recorded if a computer was in idle mode.

- *OCR system* was utilized to produce a textual representation of the content in screen capture images. We selected Tesseract (version 3.04) which is a prominent, accurate, cross platform OCR engine.
- *KE system* extracted keyphrases and named entities from the OCR-processed text. It was implemented using the Alchemy API [1].
- *OS logger* extracted the name of the active application, file path, and name of people attached to the active application as available in many messaging applications.

All extracted data were stored in a local Lucene [2] core index for high-performance indexing and retrieval.

3.2.2 Participants and Recruitment. In total, fourteen participants with varying background were recruited to take part in the experiment. Four participants quit afterward. All of the participants had different educational backgrounds, including computer scientists, business entrepreneurs, and accountants. We selected participants with higher education degrees, for they would be more likely to use their personal computers for work-related tasks, thereby allowing us to collect more and more realistic data. There were two doctoral students from a university, five with master's degrees, and three with bachelor's degrees. Participants included five females and five males with average age of 28 years ($std = 6$).

Participants were informed of their privacy upon joining the experiment and told that the data would only be stored on their local computers during the data collection phase, and only used for research purposes. In return for the effort in the experiment, participants were compensated with three movies tickets, worth around 30 euros.

The research followed the ethical guidelines established by the University of Helsinki. The research plan and an informed consent form were approved by the ethical committee of University of Helsinki and complied with the declaration of Helsinki[3] for the management of data obtained from human participants.

A consent form was signed by the users regarding the data usage policy and procedures. According to the consent some parts of the data were anonymised in connection with privacy concerns. Participants were informed that they were allowed to withdraw from the experiment at any time.

3.2.3 Procedure. Prior to starting of the digital activity monitoring, the logging software was installed on participants' laptops and was set to run continuously in the background thread for 14 days. During this period, participants were advised to perform their digital activities as usual. The participants were also asked to keep a daily diary about their digital activities. To make the diary writing convenient and avoid interference with digital tasks, we asked participants to use pen and paper to write the task entries. Participants were provided a diary template that included three fields: a brief text description about the task, related keywords describing the task, and names of people pinpointed from the task.

After briefing and software installation, participants were advised on how to write their own diaries. We intentionally advised the participants to focus on macro tasks that could consist of several activities or micro tasks. Participants were encouraged to use their own conceptualization in order to obtain realistic granularity for the tasks that emerged from participants' own understanding of what made a task meaningful. We demonstrated several examples tasks entries to ensure participants understood the requirement for the diary. For example, a task could be a complete an ongoing project, or a smaller task such as daily news reading.

After the 14-day period, the participants visited our lab. The digital activity monitoring software was uninstalled and the data stored on a designated secure computer. Participants were also encouraged to refine their diary in case they felt something was missing or incomplete. We skimmed through all written tasks and marked down all

[1] http://www.alchemyapi.com/
[2] https://lucene.apache.org/core/
[3] http://www.wma.net/en/20activities/10ethics/10helsinki/

	Screen frames	*Word occurrences*	*Tasks*
Total	49,647	4,948,591	119
Average	4,965(3,299)	494,859(363,133)	12(2)

Table 1. Descriptive statistics of the collected digital activity data from ten participants. The amount of captured screen frames, word occurrences from the OCR processing and the tasks the participants reported in their diaries.

Task description	*Detected keywords*	*People involved*
1.Emails to a potential summer trainee	research assistant, summer trainee, holidays, salary	person1
2.Processing and answering miscellaneous emails		
3.Video capture in intelligent meeting room with Theta	Ricoh, Theta, recroding pilot, 360, capture eguirectan-gutar	
4.Review of MTAP paper ANONYMIZED TITLE	face, superresolution, VLQ, real-word experiment	
5.Proactive search simulation analysis	topic, category, known item search	
6.ANONYMIZED JOURNAL NAME paper work	recommender system, recsys, cikm, experiments	
7.ANONYMIZED FUNDING AGENCY call	ANONYMIZED CALL NAME, collaboration mixed reality, eye tracking	person2, person3, person4
8.Lucene slow query issue	lucene, java, solr	
9.Review of MTAP paper: ANONYMIZED TITLE		
10.Keyword extraction from ANONYMIZED with KEA	keyword extraction, phrase, vocabulary, ontologies	person5
11.Review of ANONYMIZED TITLE...	autoencoder, skeleton, action recognition, CRWS	
12.Meeting room with pilot recording with psychophysiology	ECG, EDA, meetings record	person6, person7
13.Video capture with Theta	360, equirectgular capture	

Table 2. A sample diary from a participant. Names and titles have been anonymized to preserve privacy.

duplicated tasks. We then advised participants to consider unifying those tasks into a single task if they agreed that the tasks were duplicate. After that they individually checked and confirmed that the tasks in the final diary were in concordance to their understanding of what their main tasks were during the last 14 days.

When discussing privacy concern, one participant was uncertain about the involvement of the company data and withdrew from the experiment later that day. Another three participants withdrew during the experiment. Two participant quit after the first day. One cited frequent travels and another cited suspicions of slow computer performance caused by the OCR system. One participant withrew after two days for an unspecified reason. Complete data was gathered from ten participants.

3.2.4 Results. Table 1 presents a summary of data collected during the 14-day logging of 10 participants. The total amount of captured screens was 49,647 and the OCR process resulted in a total of 4,948,591 recorded word occurrences. Hence, the average number of screens captured and recorded word occurrences per participant was 4,965 and 494,859, respectively. The participants reported 119 tasks and there were on average 12 tasks recorded in a participant's diary. Table 2 presents a sample diary from one volunteer participant who granted permission to disclose the diary.

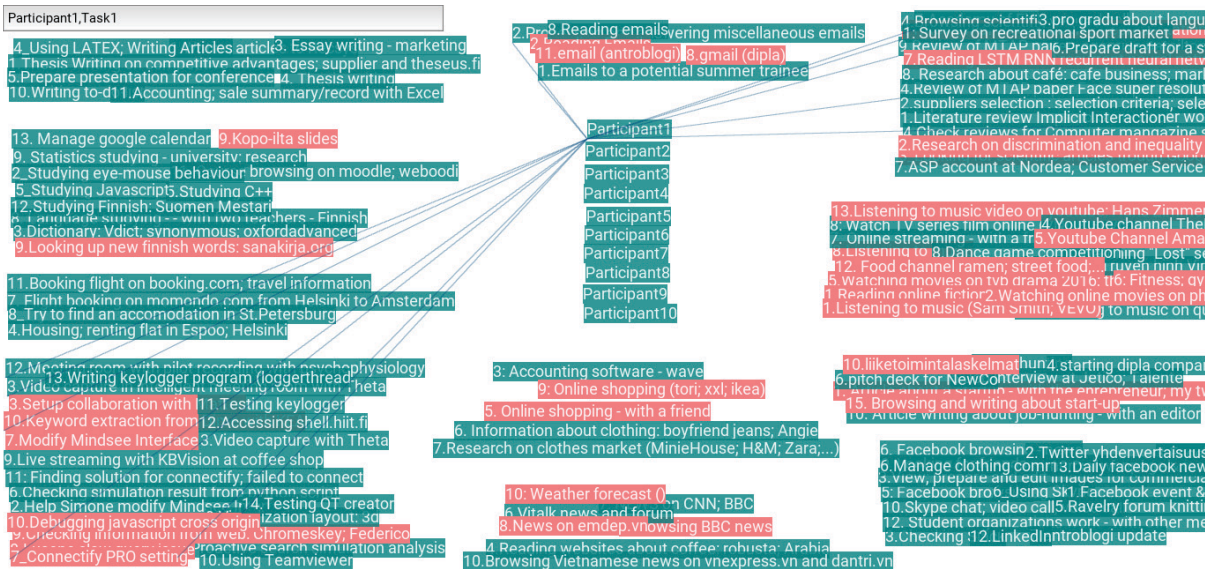


Fig. 3. The affinity diagram technique were used to categorise tasks. We used a custom made diagramming software that was programmed using Kivy version 1.9. This sofware allows the expert to interactively drag-and-drop or cluster tasks into categories. Green labeled tasks were detected tasks and red labeled tasks were incorrectly detected tasks in the experiment. Connections between tasks and a participant are represented by blue lines.

3.2.5 Task Type Classification. The diary entries were manually categorized using the affinity diagram technique. We used a custom-made software, which is shown in Figure 3. The draft version of task categorization was done by an expert. The expert analysed the semantic equivalence of textual description of diary task entries and clustered tasks into corresponding categories. The final version was made following by discussion and consensus with an external expert. The categorization resulted in 11 different task types and task distribution over 10 participants, as shown in Tables 3 and 6. This categorization allowed for the reporting of statistics on the frequency of different task types, and to further understand the accuracy of detecting different types of tasks in the unsupervised task detection phase.

3.3 Unsupervised Task Detection

The purpose of the unsupervised task detection experiment was to apply an unsupervised machine learning method to detect participants' tasks from the digital activity monitoring data. We set out to study: *How accurately can we detect user tasks from digital activity monitoring data using unsupervised learning?*

In addition, we quantified the accuracy of labeling the detected tasks and the effect of the richness of the input data when using purely raw data, keyword detection, or additional operating system data.

3.3.1 Task Detection Benchmark. Given our research goal, an unsupervised model was used to uncover the task structure in collected and OCR processed screen captures. To select a suitable model, we benchmarked three methods: Latent Semantic Analysis (LSA) [10], Latent Dirichlet Allocation (LDA) [4], and k-means clustering on Doc2Vec representation [24]. LSA learns a latent lower-dimensional representation of the input data. Each dimension in lower-dimension space can be interpreted as a task. LDA learns a generative model of term distributions over topics. LDA assigns documents into a mixture with a predefined number of topics with

Task type	*P1*	*P2*	*P3*	*P4*	*P5*	*P6*	*P7*	*P8*	*P9*	*P10*	*Task Occurrence*
Social interaction	0	0.08	0.36	0.13	0.31	0	0	0.09	0.22	0.08	0.13
Travel & accommodation	0	0	0	0	0	0.10	0	0	0.11	0.15	0.04
Writing	0	0.15	0	0.13	0.08	0.10	0.20	0	0.11	0	0.08
Learning	0	0.08	0.09	0.27	0	0.10	0	0	0.22	0.15	0.09
Research-related activity	0.36	0.15	0.09	0	0	0.30	0.20	0.09	0.11	0.15	0.15
Problem-solving	0.50	0.38	0	0	0.08	0	0.50	0.09	0.11	0.08	0.17
Industrial job-related activity	0	0	0.27	0.27	0	0	0	0	0	0.08	0.06
E-commerce	0	0	0	0.07	0.08	0.10	0	0.18	0	0	0.04
Personal Information Management	0.14	0.08	0.18	0	0	0	0.10	0	0	0	0.05
News	0	0	0	0	0.08	0.20	0	0.18	0	0.08	0.05
Video & music streaming	0	0.08	0	0.13	0.38	0.10	0	0.36	0.11	0.23	0.14

Table 3. Task type distribution over the ten participants. The *"Task Occurrence"* column indicates the mean task type distributed over participants.

	LSA	*LDA*	*Doc2Vec with K-Means*
Task detection accuracy	0.7227	0.7257	0.6927

Table 4. The overall results of the task detection accuracy with LSA, LDA, and Doc2Vec with K-means. The Wilcoxon test did not reveal statistically significant differences between the three methods.

membership probability. Each topic is represented as a task. The k-means clustering partitions the data into k disjoint clusters on top of the Doc2Vec vector space [25]. Each k-means cluster can also be interpreted as a task.

The accuracy of the task detection of the methods was measured. The ground truth of the task structure was constructed based on the descriptions of the tasks in the user's diaries by comparing the output of the method to those of the diaries. As k-means and LDA are not deterministic they were run 50 times with random seeds and the best performing run is reported.

LDA's settings relied on the number of passes and randomization seeds. The number of passes that indicated the optimal performance of probabilistic distribution over topics was 200. We consecutively examined the task detection accuracy over different runs with automatically generated randomization seeds. We report the results of the run with the highest detection accuracy. This procedure of setting parameters and accuracy assessment was done similarly for each participant.

Settings for k-means clustering were set similarly with respect to the utilization of randomization seeds to compute the model. A vector space model was firstly computed using Doc2Vec, and then the k-means algorithm was run on top of the model to produce the document clusters. In the case of both LDA and k-means, different settings showed low variance in task detection accuracy.

The results are shown in Table 4. LDA and LSA have the highest accuracy with of tasks detected, whereas Doc2Vec with k-means has the lowest detection accuracy. However, the Wilcoxon test did not reveal statistically significant differences in detection accuracy between the methods (LSA and LDA, p-value = 0.7874; LSA and Doc2Vec with k-means, p-value of 0.2317) indicating that the choice of the exact model does not have significant effect in the results.

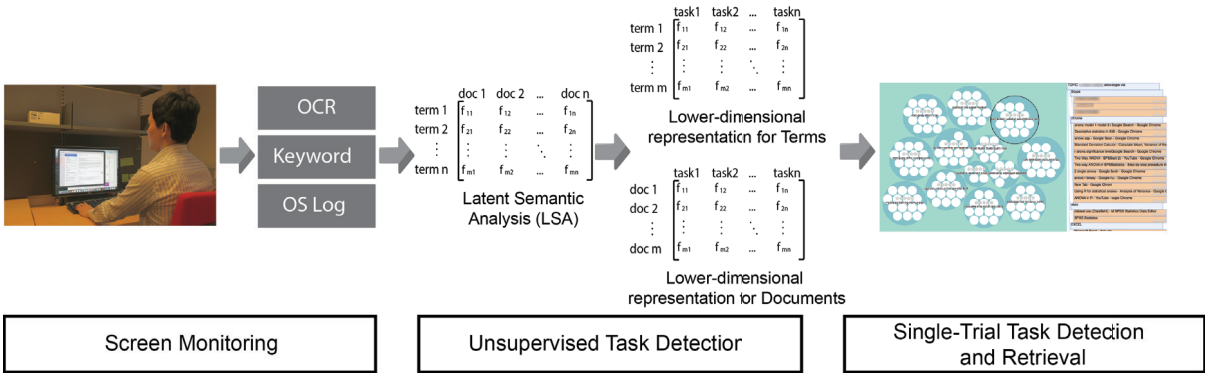


Fig. 4. The work-flow of the experiments. The digital activity monitoring data were modeled using Latent Semantic Analysis (LSA). LSA learns a latent lower-dimensional representation of the input data. Each dimension in the lower dimensional space can be interpreted as a representation of a task and used to retrieve task-relevant documents and labels.

Due to the competitive performance, lack of the need to set hyper-parameters, deterministic behavior, and the computational efficiency, we chose to utilize LSA for the remaining experiments.

3.3.2 Task Modeling. Figure 4 presents the flow of the data processing and task modeling. First, the active window on the screen is captured and saved as an image. Second, OCR software transforms the image into a text document, keywords and named entities, and application specific operating system data are extracted. Screen captures of the same document, email, or Web page were combined into a single document by comparison of their URLs and file paths. For instance, screen captures pointing to different pages of the PDF document will be combined into one document representing the whole PDF document. Third, the text, the keywords, and the operating system data are stored in a vector representing the document and the document vectors as matrix X.

LSA was then run on the matrix X. It uses Singular Value Decomposition (SVD) to decompose the original matrix to a low-rank representation:

$$\hat{X} = USV^{\mathrm{T}} \tag{1}$$

, where $\hat{X}$ is the low-rank matrix of X. We can then compute rankings for the labels and the documents from the decomposition as explained in the following sections.

To reveal the effect of different input data, such as keywords, terms, or application names, different models with varying input data were computed. The following input and data representations were configured separately to form the feature dimension of matrix X: 1) bag-of-words, 2) extracted keywords and bag-of-words, and 3) application name, extracted keywords, and bag-of-words. In all models, the occurrences of the data across the documents were normalized using tf-idf weighing [30].

The dimensionality of LSA was set individually to fit the amount of tasks reported in each participant's diary. In practice, the matrix S was set to contain only the highest Eigenvalues as specified by the number of tasks in the user's diary (rest being set to 0). While this may sound like a severe limitation of the method, the number of tasks that users' reported had fairly low standard deviation ($std = 2.0$). Moreover, in contrast to a supervised method for which the examples used to train a classifier would have to be labeled, our approach only requires one parameter to set the dimensionality of the output space.

3.3.3 Task Labeling. To visualize the task model for the user, we developed an approach to label the tasks: finding keywords that describe the dimensions in the lower-dimensional output space. Figure 5 presents an example of a detected task with visualized labels.

To select the labels, we compute a ranking for the terms by using the matrix product US, which represents the relationship between terms and tasks. Five terms with the highest values in US were selected as seed terms. These seed terms were, however, very general and not necessarily descriptive from the users' perspective. Therefore, we utilized Word2Vec to compute an embedding of keywords and terms. We then selected keywords that frequently appearing in the task and close to the highest ranked seed terms in the Word2Vec embedding space. Doing so ensured that the keywords were both related to the overall topic (close to the seed terms) and frequently appeared (relevant to the task).

3.3.4 Document Retrieval. The task modeling and labeling indicate how accurately the system can detect user activities and make them interpretable. However, this does not provide indication of the usefulness of the model for retrieving information. To measure usefulness, we built a document retrieval method that retrieves a ranked list of documents in response to a detected task. The rationale was to be able to study whether the task model can be used to proactively refind documents that could be useful resources for the user in the task context.

The vector space model of information retrieval with cosine similarity ranking was used to retrieve and rank the documents in the low-rank matrix. The input vector for the cosine similarity was the OCR processed screen capture present on the user's screen at the time of retrieval and the documents were ranked according to the cosine distance of document vectors in the low-rank matrix $\hat{X}$.

The documents were further grouped with respect to the application from which they were captured. For instance, the documents that were opened using the same PDF reader application were grouped under the application name of that PDF reader application on the user interface. Therefore, the document list on the user interface consists of two dimensions: documents that are relevant to the task, and applications that were used for the task-related activities.

3.4 Evaluation

The quality of the produced task models, the labels, and the retrieved documents were evaluated by calling participants back to the laboratory to provide a ground-truth assessment by comparing the output of the methods to their diaries.

3.4.1 System and Apparatus. To allow the participants to depict and assess the correctness of the task models, we designed an interface, which is shown in Figure 5. The interface visualizes the detected tasks by grouping entities into circles using the Zoomable Circle Packing for labels and Collapsible Indented Tree for documents as implemented in the D3.js [4] framework. This visualization was selected because it simultaneously gives both a focus-view and overview of the model, allowing for easier evaluation of focused labels and more general tasks within the model.

The interface is consisted of two views as depicted in Figure 5. The first view is the overall view of all detected tasks of a user. Each task is described using 5 task-related words. The second view is an individual view of a single task that shows extracted keywords and named entities as well as a ranked list of documents.

There are two types of circles: A big blue circle represented a detected task and smaller white circles represented keywords and named entities selected to describe the task. Every circle contained a star-rating menu in which the participants could rate the visualized information on a scale from 0 to 4. Participants rated both the overall task as a big blue circle in the overall view and the individual label describing the task as a smaller white circle in the individual task view.

[4] https://d3js.org/

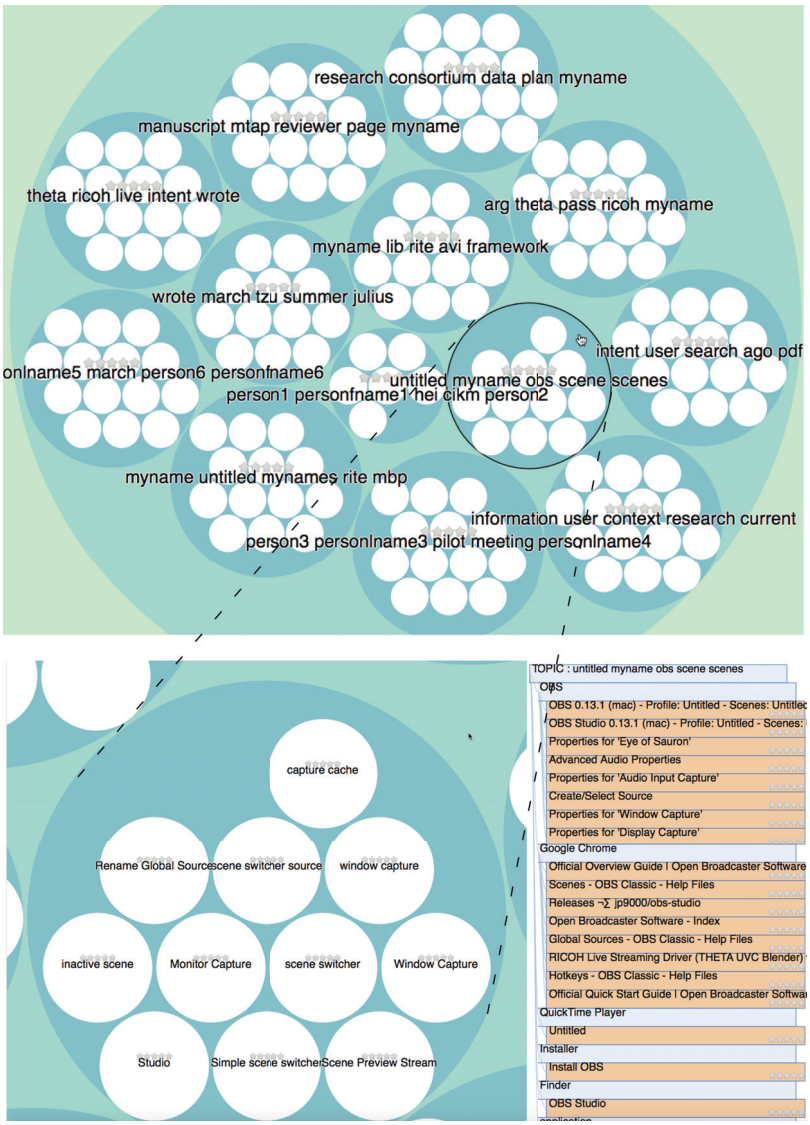


Fig. 5. A screen capture of the interface. *Top*: the task view with all detected tasks and the associated labels. Each task is visualized as a circle. *Bottom left*: view of a detected task "Video capture with Theta" for which the corresponding circle is zoomed. Inside the circle, a set of keywords and named entities extracted from screen captures describing the task. Descriptive keywords and named entities are video recording features, software menus, and tools used in the task such as "capture cache", "window capture", "inactive scene", "Rename Global Source", etc. "Window Capture" with uppercase letters refers to a name of a software and "window capture" with lowercase letters refers to text occurring as a part of a document in the running text. Therefore, they occur as separate entities in the user interface. *Bottom right*: A list of documents retrieved in response to the detected task. The list contains documents from variety of applications and systems such as, OBS software where new scenes or video projects were created, Web browsers by means of which the user looked up Theta-related tutorials and OBS's help information, Quick Time Player to re-play recorded videos, Finder through which the user explore location of OBS application, and so on.

	Document retrieval	*Labels*
NDCG	0.92	0.72
Precision@1	0.93	0.80
Precision@5	0.93	0.70
Precision@10	0.92	0.70
Precision@20	0.89	

Table 5. The results of the document retrieval and labeling as precision at N and NDCG.

The individual task view is accessible by selecting a task in the overall view. After selecting a task by clicking a big blue circle in the overall view, the system retrieved a ranked list of documents and displayed them on the right side of the interface as shown in Figure 5. Each document contained a star-rating menu similar to the circle. Participants also had to rate every document in the list. To return to the overall view and continue rating the rest of the tasks, participants simply click anywhere outside the big blue circle on the individual task view.

Participants were shown the interface on an 24-inch LCD monitor. They could operate the interface by zooming in on, or by switching alternatively between the first view and second view using a mouse and a keyboard.

3.4.2 Procedure. Participants were first briefed regarding the purpose of the experiment: detecting tasks and assessing the relevance of tasks, labels, and documents. Before the beginning of the experiment, we ran the system to compute the task model from the participant's own digital activity monitoring data, which were presented on the screen (see Figure 5). The participants then rated each of the elements on the screen in comparison to the information in their own diary. The participants had an unlimited time to complete the assessments.

The task assessments criterion was *task accuracy* and *relevance* relative to a subjective evaluation of the quality of the system produced tasks, labels, and documents with respect to the actual tasks in their diaries.

We applied two levels of assessment:

- Accuracy: participants were asked to explicitly pointed out either (0) no tasks matched for a task, and for items (keywords and documents), those that do not belong to a corresponding task. Otherwise, above (0) indicates correctly formulation of a task and, for items (keywords and documents), those that belong to a corresponding task. This provided the ground-truth that was used in the evaluation of the accuracy of digital task retrieval.
- Relevance of the content of the task - the following scale was used: (1) slightly relevant; (2) moderately relevant; (3) highly relevant; and (4) absolutely relevant. This was used in the evaluation of the relevance of task related information.

3.4.3 Measures. Two standard measures were used to characterize the meaningfulness of the task models: mean score and accuracy. The score was computed simply as an average rating that the users gave for the task description. The accuracy was computed as the binarized output characterizing a standard multi-class classification performance of the model. Scores equal to or greater than one were marked accurate and the degree of relevance, and scores of 0 were marked inaccurate. While this is a rough binarization, when used in conjunction with a graded mean score, it provides an objective view of the output of the models.

Similarly, two standard measures were used to characterize the meaningfulness of the labels and retrieved documents: precision at cut-off levels and normalized discounted cumulative gain [18].

3.4.4 Task Detection Results. Table 4 shows a summary of the overall results of task detection. The participants assigned a mean score of (2.754/4, std=0.96) for the tasks that were detected and displayed to them, which

Task type	*Terms*	*Keywords*	*Terms & keywords*	*Keywords & application names*	*Terms & application names*	*Terms & application names & keywords*
Social interaction	**1**	0.93	0.80	0.53	0.73	0.73
Travel & accommodation	**1**	0.75	0.75	0.50	0.50	0.75
Writing	**1**	0.78	0.56	**1**	0.78	0.67
Learning	**0.82**	0.64	0.64	0.73	0.64	0.55
Research-related activity	**0.76**	0.71	0.71	0.47	0.59	0.71
Problem-solving	0.67	**0.76**	0.67	0.67	0.62	0.71
Industrial job-related activity	0.63	0.63	0.88	0.75	0.88	**1**
E-commerce	0.60	0.40	0.80	0.40	**1**	**1**
Personal information management	0.50	0.67	**0.83**	0.50	0.67	**0.83**
News	0.50	**0.83**	0.67	0.67	0.67	0.67
Video & music streaming	0.47	0.53	0.53	**0.71**	0.65	0.65
Overall	**0.72**	0.71	0.69	0.64	0.68	0.72

Table 6. Accuracy of task detection with respect to different models with varying input data and for different task types. The highest values for each type are in bold face. Models indicate large variance with respect to model and task type. The simplest term input with bag-of-words representation has the highest overall accuracy.

according to our scale indicates relevant to highly relevant task detection. The corresponding accuracy was 72.27%.

During the experiment we also noticed that some tasks were assigned a high score, but the participants indicated that they were not in their original diary. In other words, the system had detected tasks that were meaningful for the users, though they were not manually entered as tasks in the participants' diaries. This allowed us to compute the precision for task detection, which was higher than the accuracy, for there were additional tasks in the ground-truth pool. The precision was found to be 76.85%.

An example of a task that was recognized by the system, but was missing from a participant's diary was a troubleshooting network issue with Wi-Fi software that required the participant's attention for several days. This participant had to spend some time fixing the software during the 2-week digital activity monitoring, but had forgot to specify this in the diary.

3.4.5 Task Labeling Results. The "labels" column in Table 5 shows a summary of the results of the task labeling. The NDCG and precision values were computed for the top-10 labels retrieved for the task as in general more than ten labels would not be useful for recognizing a task, but rather cluttering. NDCG for the labels was 0.72 and Precision for the first label was 0.8 indicating that sensible labels were retrieved for the tasks and that in over 80% of the cases, users could recognize the tasks from the first label.

3.4.6 Document Retrieval Results. The "document retrieval" column in Table 5 shows the results of the quality of retrieved documents. The NDCG and precision values were computed for the top-20 documents. NDCG for document retrieval was 0.92 and precision of around 0.9 was stable for the list of the top-20 documents, shown to users in the experiment. This indicates that most of the retrieved documents were found to be highly relevant for the task.

In general, participants seemed to be more satisfied with the set of retrieved documents than they were with the labels. Qualitative feedback from the participants suggested that in some cases the labels were somewhat recognizable but were not considered high-quality names that the participants would use themselves to label the

Task type	*Score*
Social interaction	3.33
Travel & accommodation	2.50
Writing	2.89
Learning	2.09
Research-related activity	2.70
Problem solving	1.90
Industrial job-related activity	1.88
E-commerce	2.20
Personal information management	1.33
News	1.50
Video & music streaming	1.65

Table 7. Descriptive statistics of the tasks and users with respect to different task types. Mean score indicates the meaningfulness of the detected tasks with respect to task types.

task. For instance, based on term frequency, several keywords could be extracted from trivial sources such a tool name on a software panel in "accounting software - Wave", commercial ads that kept appearing in social network page such as "browsing Facebook", or keywords from recommended videos on Web pages in "online streaming with friends". While these were associated with certain tasks and users could recognize the task, they were by no means labels that the users would have personally used to characterize these tasks.

3.4.7 Modeling with different data sources. In order to understand which type of input to the unsupervised model was the most useful and whether there was a variance in the kind of input that was useful for detecting a particular type of task, we also constructed and measured the performance of different models with varying input. In total, six different models with different input were trained, and results are shown in Table 6 and 7: *"terms"* as bag-of-words, *"keywords"*, *"terms & keywords"*, *"keywords & application names"*, *"terms & application names"*, and *"terms & keywords & application names"*.

As ground truth assessments from the users were only available for the *"terms"* input model and obtaining assessments from the users was not possible due to the large pool of model outputs, we ran an expert annotation to obtain relevance assessments for the output of each model. The diary entries and assessments for the *"terms"* model were taken into account and the expert assessed binary relevant / irrelevant labels for a task outputted by any of the models. The assessment was binary because the external expert was not able to provide a graded assessment, but found it plausible to assess whether the model output was relevant and associated with one of the tasks or was irrelevant and not associated with a task. A two-way ANOVA was run with task type and model as explanatory variables and accuracy as a response variable. The test did not reveal significant differences between the different models.

This was a surprising result as it implies that even the simple model trained with only bag-of-words vectors performed equally as well as the others. This indicates that the data were rich enough to lead to high accuracy with a simple model and without any pre-processing. However, large variances of accuracy were observed between tasks and models depending on data sources.

The *"term"* model was successful in five task types which are *"Social interaction"*, *"Travel & accommodation"*, *"Writing"*, *"Learning"*, and *"Research-related activity"*. Whereas, the *"keyword"* model achieved higher accuracy in the two task types of *"Problem-solving"* and *"News"*. The possible explanation is that repetitive occurrence of keywords in the screen captures gave better clue to the model to detect these tasks.

Moreover, in the task type *"Personal information management"*, adding more features from *"keyword"* and *"application name"* to the *"term"* model produced higher detection accuracy. For *"E-commerce"* and *"Video & music streaming"*, it showed less effective to utilize only either *"term"* or *"keyword"* input, as these task types are comprised of screen captures with less text but more visual information such as, from videos and pictures. They are also reliant on a few number of applications, and adding *"application name"* feature gives strong clue to the model.

The results also show that using both terms and keywords can lead to lower overall accuracy than using a model with only terms or only keywords. In some rare cases, the terms and the keywords are partially overlapping and the feature is duplicated in the representation of the task. A false positive that is duplicated can accumulate the error and lead to reduced accuracy. However, such differences were not found to be statistically significant or to have variance across participants and tasks. Consequently, there is no generalizable difference between these models.

4 SINGLE-TRIAL TASK DETECTION AND RETRIEVAL EXPERIMENT

The purpose of the single-trial experiment was to study the usefulness of the resulting task model in a single-trial real-time task detection and information retrieval scenarios. To put it simply, it tested whether the model could correctly classify unseen input resulting from user interactions to a task in the task model, and proactively retrieve relevant related information.

4.1 Participants and Apparatus

The same participants who participated in the previously described experiments were called back to the laboratory one week after the unsupervised task detection experiment. They provided an assessment of the quality of retrieval in on-line interactive settings. Similar apparatus and relevance assessment were applied, through additional assessments for the proactive retrieval of task-relevant information.

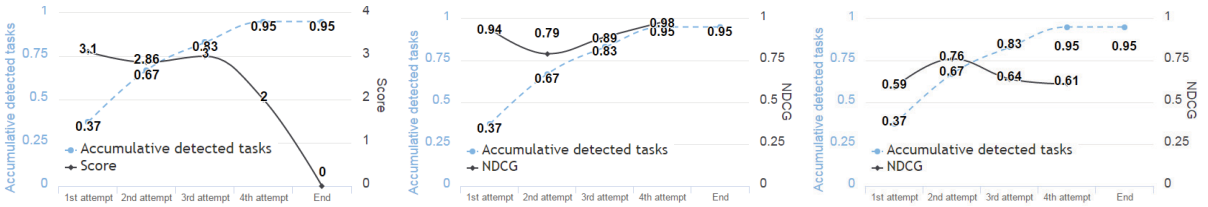
4.2 Procedure

Prior to the actual experiment, we asked the participants to select from their diaries six tasks on which they were still actively working or on which that they had worked most recently. The participants then used a computer running the digital activity monitoring system to perform activities related to the selected task one at a time. Participants were explicitly advised to continue their tasks (i.e. to perform new activities dedicated to the chosen task).

The output of the digital activity monitoring system was fed into the LSA model, which resulted in the prediction of the task that the participant was performing. As no significant differences were found in the overall performance of the different models in the unsupervised task detection experiment, we used the "terms" model with only bag-of-words representation.

The predicted task was visualized for the user in a visualization shown in Figure 5. When the system detected a task it zoomed in to the circle representing that task and proactively retrieved the related documents from the digital activity database.

The participants' task was interrupted at 30-second intervals to obtain a relevance assessment on the task detection and retrieval. In other words, every 30 seconds (up to 120 seconds), we asked the participant to look at the task detection system and to assess the relevance of the task on which the system zoomed in and if the task was detected correctly, we also asked the participants to assess the labels and documents. If the task was not detected after 4 attempts (120 seconds), the task was marked as failed and the participant started the next trial with the next task.



(a) Task detection accuracy over time (b) Document retrieval NDCG over time (c) Labeling NDCG over time

Fig. 6. Results of the single-trial task detection and retrieval. The figures present different performance measures (y-axis) with respect to the elapsed task time (x-axis). The blue dashed line indicates the cumulative accuracy of detected tasks. The black line indicates the performance measure.

4.3 Measures

Similarly to the unsupervised task detection experiment, the main evaluation criterion was the quality score that the participants provided. We also used Normalized Discounted cumulative gain (NDCG) and precision at N to measure the performance of the information retrieval performance of documents and labels. All measures were computed at every interruption point (at 30, 60, 90, and 120 seconds).

4.4 Results

The overall result after all trials and attempts (mean over the tasks and participants at 120 seconds) shows a task detection accuracy of 95%. More precisely, 57 out of 60 tasks (6 tasks per participant, 10 participants) were correctly detected. There were 3 undetected tasks: One was a task looking up words on a dictionary website, the second was a writing task in which the participant constantly wrote an email message concerning ”mail to ANONYMIZED-ORGANIZATION about pilot meeting”, and the third was a researching task including browsing several new Web pages related to ”Article about a startup company”.

The temporal results at the task interruption points, are shown in Figure 6a. After the first interruption point the system detected the task at an accuracy of 37% and after the second interruption (at 60 seconds) at an accuracy of 67%. After the third attempt, 83% of tasks were detected correctly. The results indicate that although a majority of tasks can be detected within one minute after the user starts to interact with the computer, the best detection accuracy is achieved only after two minutes of digital activity monitoring.

The temporal graphs also show that the subjective quality score that the users provided was high for the first three attempts (3.18-3.31/4), but dropped at the fourth attempt due to the tasks becoming harder to detect. This indicates an expected tradeoff of a majority of the tasks being easy to detect even in a single-trial setup, and a small portion of the tasks being very difficult to detect. Figure 6b presents the NDCG of document retrieval with respect to the attempts. The NDCG was high throughout the attempts, varying between 0.98 and 0.79 with a slightly lower value at the second attempt. It is notable that both the task detection quality score that the participants provided and the NDCG were already high starting from the first attempt indicating that when the tasks were detected correctly and the document retrieval also worked with real-time streaming input.

Table 8 shows a summary of all measures for the document retrieval. The results show a constant high precision and NDCG for the top 20 retrieved documents throughout the attempts, with NDCG over 0.9 in the first attempt (after 30 seconds of digital activity monitoring). We observed a slight drop at the second attempt. This indicates that there were some documents that were harder to retrieve and required more evidence for the task model to converge to the correct task and to the improved document ranking. Similarly, Table 9 presents all measures

Attempt	*1*	*2*	*3*	*4*
Score	3.31(0.61)	3.21 (0.67)	3.18 (0.79)	2.67 (0.98)
NDCG	0.94	0.79	0.89	0.98
P@1	1	0.78	0.9	1
P@10	0.95	0.82	0.87	0.92
P@20	0.93	0.89	0.87	0.83

Table 8. Document precision at 1, 10, and 20 in the single-trial task detection and document retrieval experiment. Results are reported with respect to attempts (task interruption at 30 seconds intervals).

Attempt	*1*	*2*	*3*	*4*
Score	1.20(0.67)	1.91 (0.67)	1.66 (1.01)	1.63 (0.73)
NDCG	0.59	0.76	0.64	0.61
P@1	0.59	0.89	0.80	0.86
P@5	0.60	0.72	0.64	0.60
P@10	0.52	0.62	0.63	0.60

Table 9. Labeling precision at 1, 5, and 10 in the single-trial task detection and document retrieval experiment. Results are reported with respect to attempts (task interruption at 30 seconds intervals).

for task labeling. The values are generally low for labeling indicating that labeling tasks remains challenging compared to retrieving task-relevant documents.

5 DISCUSSION

5.1 Summary of results

The aim of this research was to investigate the modeling of digital tasks from screen captures for task-aware information retrieval. We explored two different questions in each of two experiments. *How accurately can we detect user tasks from digital activity monitoring data using unsupervised learning?* The unsupervised task detection experiment showed that it is possible to detect participants' tasks by only monitoring their screens via screen capture monitoring. The unsupervised learning method, trained on the digital activity monitoring data, detected tasks with an accuracy of 72.27%. Including richer input, such as named entities or application names, was not found to be helpful in improving task detection accuracy, but high variance between the model accuracy was observed between some task types. Some task types were reliant on named entities or application names. Moreover, the accuracy of fusing both bag-of-words and named entities feature was lower than either using only one of them. The future work will investigate the task types and the reasons that could cause this reduced accuracy.

In this article, we did not compare the retrieval results to other ranking methods as the focus of the research was to study what can be learned from the type of data we collect. In addition, the ad-hoc retrieval was shown to be less sensitive to the exact ranking method used [1]. Nevertheless, the results still show high user satisfaction and retrieval effectiveness by utilization of LSA with only bag-of-words data. Future work could address effects of different computational methods and data input on the task detection accuracy, for example, by using temporal or other background information.

The second experiment on single-trial task detection experiment showed that, given a task had been detected with success (demonstrated with over 72% accuracy in the first experiment) participants' tasks were detected at

95% of accuracy from unseen interactions. The majority of tasks were detected after two attempts within one minute from the beginning of the digital activity monitoring.

Over two weeks, average screen captures yielded per participant 4,965 screen shots and 494,859 word occurrences per participant. The participants reported 119 tasks in their diaries and there were on average 12 tasks recorded in a participant's diary. This shows that, while participants performed a lot of digital activities and micro tasks, they are typically connected to fairly few macro tasks that are from a routine and are meaningful for the participants.

How accurately can we use the resulting task model for on-line task detection and proactive retrieval of task-relevant information? In both experiments the document retrieval was successful, with an accuracy of over 90%. In the first experiment the NDCG for document retrieval was 0.92, and precision of around 0.9 was stable for the list of the top-20 documents, indicating that most of the retrieved documents were found to be highly relevant for the task.

In the second experiment the NDCG of model-based document retrieval was also over 0.9 throughout indicating high document retrieval effectiveness. In both experiments, however, the task labeling effectiveness was more challenging, and participants seemed to be more satisfied with the set of retrieved documents than they were with the labels.

5.2 Implications

The implications of the results for user modeling are striking because they open opportunities to learn user models from a "single source" across the confines of individual applications. Our results demonstrate that it is possible to create comprehensive, accurate, and robust user models from simple input just by watching the screen.

While this may sound like a limitation compared to more structured user signals, such as clicks or direct input within an application, it enables general user modeling that can be used across applications, utilizing data from one application in another solving issues such as cold start problems [36].

Our results are extremely promising for user modeling goals, but they also evoke concern for user privacy. While this concern is by no means new (see, for example, [34]), the current single-login mechanisms already collect a variety of traces of human behaviour on the World Wide Web, and many of our emails are stored and hosted by external service providers. Human digital traces may soon provide enough data for high-accuracy task detection and may reveal and reflect our interests and activities much more accurately than we expected.

The resulting sensitive information could be potentially exploited in unethical ways or used against users' interests. To this end, another important opportunity derived from our approach is being able to learn user models from an end-user device (or front end). This does not require access to the data structure of the service provider or application developer, and the data obtained can be owned and utilized by the actual user given appropriate platforms and tools. Echoing these ideas are recent development in data privacy that promote human-centred personal data management and processing, putting the users at the center and in control of their own data [8].

Paradoxically, allowing extreme digital activity monitoring seems to lead to unpredictably reliable models of users' tasks and interests, while allowing fair information practices that may preserve the ownership of personal data in the hands of users.

5.3 Limitations

The specific difficulty encountered with the introduction of the digital activity monitoring method was privacy. Participants were fully aware of their activities being monitored, and thus they may have concealed some of their behavior on purpose. This limitation was predictable. Our expectation was that participants could reveal most of their activities that they considered to be less private. Besides, there was no negative impact on the results of

our method as what we planned to achieve was to investigate the possibility of making inferences about users' digital tasks with the given behavioral data collected from digital activity monitoring.

Another concern was the resolution of the task output. Our validation method relied on written diaries describing the subjective relevance of the tasks from the participants perspective as ground-truth. This may have constrained the task model. Moreover, while the 14-day digital activity monitoring of ten participants resulted in large data, consisting of nearly 50,000 screen frames, it was a fairly small sample compared to what could be sourced from longer term instrumentation with a larger population. Additionally, the average amount of tasks among participants was also small and did not vary significantly (average of 12 tasks per person). This could be extended in the future work by utilizing hierarchical modeling technique that could automatically fit the number of tasks to a larger variety of micro and macro tasks.

Lastly, our aim was to build a user model without human supervision about tasks that are meaningful to the users from just a single source of data. Therefore, the models were fixed based on diary entries and did not account for temporal changes as well as chronological order between activities. In the present experiments, we did not intend to detect task switches in the single-trial single-task detection experiment. However, this is potentially an important feature of a model when considering deployment in real-life settings. Future work could also investigate multi-view models that would be able to model the variance in the different types of data extracted, such as named entities, keywords, or application names, and fit to different granularity of tasks.

6 CONCLUSIONS

In this paper, we have exploited 24/7 digital activity monitoring to detect tasks from natural computer usage. We conducted two experiments from unsupervised task detection from long-term digital activity data, and built an on-line system to proactively retrieve real-time task-relevant information corresponding to streaming digital activity input. The experimental results show that an unsupervised method can detect these tasks with a high accuracy and retrieve task-relevant information automatically using the task model. Within the scope of tasks that participants' reported, the unsupervised modeling of digital activity data was performed at nearly human level. This result was also found to hold in a single-trial scenario in which model input was not observed in the training phase and task detection was performed on-line using streaming digital activity input. Surprisingly, the simplest model with only bag-of-words data representation was most effective suggesting that the richness of data available from digital activity monitoring seems to be the key to detecting diverse and subtle human interests.

REFERENCES

[1] Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. 2009. Improvements That Don'T Add Up: Ad-hoc Retrieval Results Since 1998. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM '09)*. ACM, New York, NY, USA, 601–610. https://doi.org/10.1145/1645953.1646031

[2] N. J. Belkin, R. N. Oddy, and H. M. Brooks. 1982. Ask for Information Retrieval: Part I.: Background and Theory. *Journal of Documentation* 38, 2 (1982), 61–71. https://doi.org/10.1108/eb026722

[3] Daniel Billsus and Michael J. Pazzani. 2000. User Modeling for Adaptive News Access. *User Modeling and User-Adapted Interaction* 10, 2 (2000), 147–180. https://doi.org/10.1023/A:1026501525781

[4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* 3 (March 2003), 993–1022. http://dl.acm.org/citation.cfm?id=944919.944937

[5] Pam Briggs, Elizabeth Churchill, Mark Levine, James Nicholson, Gary W. Pritchard, and Patrick Olivier. 2016. Everyday Surveillance. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 3566–3573. https://doi.org/10.1145/2851581.2856493

[6] Katriina Byström and Preben Hansen. 2005. Conceptual Framework for Tasks in Information Studies. *J. Am. Soc. Inf. Sci. Technol.* 56, 10 (Aug. 2005), 1050–1061. https://doi.org/10.1002/asi.v56:10

[7] Sergey Chernov. 2008. Task Detection for Activity-based Desktop Search. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08)*. ACM, New York, NY, USA, 894–894. https://doi.org/10.

1145/1390334.1390569

[8] Eun Kyoung Choe, Nicole B. Lee, Bongshin Lee, Wanda Pratt, and Julie A. Kientz. 2014. Understanding Quantified-selfers' Practices in Collecting and Exploring Personal Data. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 1143–1152. https://doi.org/10.1145/2556288.2557372

[9] Elizabeth F. Churchill. 2014. Scrupulous, Scrutable, and Sumptuous: Personal Data Futures. *interactions* 21, 5 (Sept. 2014), 20–21. https://doi.org/10.1145/2656856

[10] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science* 41, 6 (1990), 391–407. https://doi.org/10.1002/(SICI)1097-4571(199009)41:6⟨391::AID-ASI1⟩3.0.CO;2-9

[11] Carsten Eickhoff, Jaime Teevan, Ryen White, and Susan Dumais. 2014. Lessons from the Journey: A Query Log Analysis of Within-session Learning. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM '14)*. ACM, New York, NY, USA, 223–232. https://doi.org/10.1145/2556195.2556217

[12] Gerhard Fischer. 2001. User Modeling in Human–Computer Interaction. *User Modeling and User-Adapted Interaction* 11, 1-2 (March 2001), 65–86. https://doi.org/10.1023/A:1011145532042

[13] Ramanathan Guha, Vineet Gupta, Vivek Raghunathan, and Ramakrishnan Srikant. 2015. User Modeling for a Personal Assistant. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM '15)*. ACM, New York, NY, USA, 275–284. https://doi.org/10.1145/2684822.2685309

[14] Kirstie Hawkey and Kori M. Inkpen. 2006. Keeping Up Appearances: Understanding the Dimensions of Incidental Information Privacy. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 821–830. https://doi.org/10.1145/1124772.1124893

[15] Eric Horvitz, Jack Breese, David Heckerman, David Hovel, and Koos Rommelse. 1998. The LumièRe Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI'98)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 256–265. http://dl.acm.org/citation.cfm?id=2074094.2074124

[16] Wen Hua, Yangqiu Song, Haixun Wang, and Xiaofang Zhou. 2013. Identifying Users' Topical Tasks in Web Search. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining (WSDM '13)*. ACM, New York, NY, USA, 93–102. https://doi.org/10.1145/2433396.2433410

[17] Peter Ingwersen and Kalervo Järvelin. 2005. *The Turn: Integration of Information Seeking and Retrieval in Context (The Information Retrieval Series)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[18] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated Gain-based Evaluation of IR Techniques. *ACM Trans. Inf. Syst.* 20, 4 (Oct. 2002), 422–446. https://doi.org/10.1145/582415.582418

[19] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2005. Accurately Interpreting Clickthrough Data As Implicit Feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '05)*. ACM, New York, NY, USA, 154–161. https://doi.org/10.1145/1076034.1076063

[20] Eunju Kim, Sumi Helal, and Diane Cook. 2010. Human Activity Recognition and Pattern Discovery. *IEEE Pervasive Computing* 9, 1 (Jan. 2010), 48–53. https://doi.org/10.1109/MPRV.2010.7

[21] Alfred Kobsa. 2007. *Generic User Modeling Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 136–154. https://doi.org/10.1007/978-3-540-72079-9_4

[22] Claudio Lucchese, Salvatore Orlando, Raffaele Perego, Fabrizio Silvestri, and Gabriele Tolomei. 2011. Identifying Task-based Sessions in Search Engine Query Logs. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (WSDM '11)*. ACM, New York, NY, USA, 277–286. https://doi.org/10.1145/1935826.1935875

[23] Claudio Lucchese, Salvatore Orlando, Raffaele Perego, Fabrizio Silvestri, and Gabriele Tolomei. 2013. Discovering Tasks from Search Engine Query Logs. *ACM Trans. Inf. Syst.* 31, 3, Article 14 (Aug. 2013), 43 pages. https://doi.org/10.1145/2493175.2493179

[24] J. MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*. University of California Press, Berkeley, Calif., 281–297. http://projecteuclid.org/euclid.bsmsp/1200512992

[25] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781

[26] Feng Qiu and Junghoo Cho. 2006. Automatic Identification of User Interest for Personalized Search. In *Proceedings of the 15th International Conference on World Wide Web (WWW '06)*. ACM, New York, NY, USA, 727–736. https://doi.org/10.1145/1135777.1135883

[27] Andreas S. Rath, Didier Devaurs, and Stefanie N. Lindstaedt. 2009. UICO: An Ontology-based User Interaction Context Model for Automatic Task Detection on the Computer Desktop. In *Proceedings of the 1st Workshop on Context, Information and Ontologies (CIAO '09)*. ACM, New York, NY, USA, Article 8, 10 pages. https://doi.org/10.1145/1552262.1552270

[28] Xuehua Shen, Bin Tan, and ChengXiang Zhai. 2005. Context-sensitive Information Retrieval Using Implicit Feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '05)*. ACM, New York, NY, USA, 43–50. https://doi.org/10.1145/1076034.1076045

[29] Fabrizio Silvestri. 2010. Mining Query Logs: Turning Search Usage Data into Knowledge. *Found. Trends Inf. Retr.* 4, 1—2 (Jan. 2010), 1–174. https://doi.org/10.1561/1500000013

[30] Karen Sparck Jones. 1988. Document Retrieval Systems. Taylor Graham Publishing, London, UK, UK, Chapter A Statistical Interpretation of Term Specificity and Its Application in Retrieval, 132–142. http://dl.acm.org/citation.cfm?id=106765.106782

[31] Bin Tan, Xuehua Shen, and ChengXiang Zhai. 2006. Mining Long-term Search History to Improve Search Accuracy. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '06)*. ACM, New York, NY, USA, 718–723. https://doi.org/10.1145/1150402.1150493

[32] Jaime Teevan, Susan T. Dumais, and Eric Horvitz. 2005. Personalizing Search via Automated Analysis of Interests and Activities. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '05)*. ACM, New York, NY, USA, 449–456. https://doi.org/10.1145/1076034.1076111

[33] Jaime Teevan, Susan T. Dumais, and Eric Horvitz. 2010. Potential for Personalization. *ACM Trans. Comput.-Hum. Interact.* 17, 1, Article 4 (April 2010), 31 pages. https://doi.org/10.1145/1721831.1721835

[34] Eran Toch, Yang Wang, and Lorrie Faith Cranor. 2012. Personalization and privacy: a survey of privacy risks and remedies in personalization-based systems. *User Modeling and User-Adapted Interaction* 22, 1 (2012), 203–220. https://doi.org/10.1007/s11257-011-9110-z

[35] Manisha Verma and Emine Yilmaz. 2014. Entity Oriented Task Extraction from Query Logs. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management (CIKM '14)*. ACM, New York, NY, USA, 1975–1978. https://doi.org/10.1145/2661829.2662076

[36] Chirayu Wongchokprasitti, Jaakko Peltonen, Tuukka Ruotsalo, Payel Bandyopadhyay, Giulio Jacucci, and Peter Brusilovsky. 2015. *User Model in a Box: Cross-System User Model Transfer for Resolving Cold Start Problems.* Springer International Publishing, Cham, 289–301. https://doi.org/10.1007/978-3-319-20267-9_24

[37] Dingqi Yang, Daqing Zhang, Longbiao Chen, and Bingqing Qu. 2015. NationTelescope: Monitoring and visualizing large-scale collective behavior in {LBSNs}. *Journal of Network and Computer Applications* 55 (2015), 170 – 180. https://doi.org/10.1016/j.jnca.2015.05.010

[38] Zack Zhu, Ulf Blanke, Alberto Calatroni, and Gerhard Tröster. 2013. Human Activity Recognition Using Social Media Data. In *Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia (MUM '13)*. ACM, New York, NY, USA, Article 21, 10 pages. https://doi.org/10.1145/2541831.2541852

Received February 2017; revised May 2017; accepted July 2017